

Applepen.

23 марта 2019 г.

1 Постановка задачи.

Существует торговая сеть Applepen, продающая яблоки и карандаши, в которую входят несколько магазинов. Имеется информация о закупках (поставки яблок и карандашей два раза в месяц), продажах (лог транзакций, по записи на каждую проданную позицию), инвентаре (месячные данные общего количества яблок и карандашей на складе). Каждый магазин собирает информацию о закупках, продажах, инвентаре. Данные отсортированы по дате и доступны в формате CSV. Необходимо получить следующие данные в CSV-файлах: состояние склада на каждый день, месячные данные о количестве сворованного товара, агрегированные данные об объемах продаж и количестве сворованной продукции по штату и году.

2 Решение задачи.

Чтобы выполнить задание нам понадобится библиотека pandas. Чтобы получить необходимые данные для каждого склада, создадим функцию csv-dist-reader. Чтобы считать входные данные нам понадобится функция read_csv. Построим эту функцию: В reader поместим данные о продажах на складе, в DF-inventory-данные об инвентаре, в DF-supply-данные о поставках на склад. Зададим переменные current-apple, current-pen, sold-apple, sold-pen, supply-apple, supply-pen, stolen-apple, stolen-pen, обозначающие количество проданных товаров, поставленных товаров, инвентаря. Присвоим им первоначально нулевые значения. Создадим 2 объекта типа dataframe: DF-daily, DF-stolen со столбцами "data", "apple", "pen" с помощью функции Dataframe из библиотеки pandas. Эти объекты содержат данные о состоянии склада на каждый день и ежемесячном количестве сворованного товара соответственно. Создадим переменную, обозначающую предыдущую дату prev-date. Присвоим ей первоначальное значение "0000-00-00". Также введём переменные daily-index, supply-index, stolen-index, inventory-index, присвоив им начальные нулевые значения. С помощью цикла for пройдем по всем записям о продажах. Если дата совпадает с датой предыдущей продажи, то увеличим количество проданных товаров: яблок или карандашей. В зависимости от того "a" или "p" написано в описании транзакции, мы увеличим sold-apple или sold-pen на 1. В ином случае, если новая продажа происходит на другой день и если предыдущий день был первым или пятнадцатым в месяце, то увеличиваем переменные, обозначающие количество поставленного товара. Supply-apple и supply-pen увеличим на значение элемента

номера supply-index в столбцах apple и pen DF-supply соответственно. После этого увеличим supply-index на 1. Меняем переменные current-apple и current-pen, обозначающие количество товаров в данный момент на складе, прибавляя к нему количество поставленного товара и вычитая количество проданного. После этого проверяем является ли текущая дата началом нового месяца. Если да, то вычисляем количество сворованного из магазина товара, содержащееся в переменных stolen-apple и stolen-pen. Вычисляем эти переменные, вычитая из текущего их количества количество инвентаря, содержащееся в строке под номером inventory-index DF-inventory. После этого изменим текущее количество товара присвоив переменным значение соответствующего инвентаря. После этого добавляем в DF-daily количество товара в текущий день и в DF-stolen количество украденного товара. Теперь после того как цикл закончится у нас в DF-daily будет содержаться информация о состоянии склада на каждый день, а в DF-stolen информация о месячных данные о количестве сворованного товара с этого склада. Что нам и было нужно. Применяя построенную функцию csv-dist-reader для каждого склада мы получим необходимый результат. Наша задача выполнена.

Программы Python 3.6 Jupiter Notebook

Авторы Екатерина Ворончихина Таипов Михаил Виль Вадим